An Analysis of Personalized Learning Platform Model

by

Golam Rabbi
20101086

A project submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
October 2024

Declaration

It is hereby declared that

1. The project submitted is my/our own original work while completing degree at Brac University.

2. The project does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The project does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Golam Rabbi

20101086

Approval

The project titled "An Analysis of Personalized Learning Platform Model" submitted by

1. Golam Rabbi (20101086)

Of Summer, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October 23, 2024.

Examining Committee:

Supervisor:
(Member)

Arif Shakil
Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam,PhD
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi,PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

In the modern continuously developing field of education, it is concluded that the establishment of individual learning appears to be possible with the help of machine learning solutions. The case being presented in this paper calls for the implementation of a concept of a learning platform that is packaged with a state-of-the-art machine learning tool set to boost academic achievement. The proposed method consists of three main components: This paper presents a seven-feature approach that includes detailed response and feedback, dynamic control of learning processes, and a recommendation system. The recommendation system applies information, demographic and collaborative information about learning resources to each learner based on individual learning style and academic accomplishment records.

Adaptive learning thus self-organizes content based on the various aspects of student interaction and achievement so that a perfect learning path is achieved. Feedback as motivation and constant improvement, feedback for timely, useful, and individualized criticism through sentiment and Natural language processing. The integration of these components, however, suggests the potential for developing the present recommendation platform to offer a more productive and enjoyable educational experience that meets the needs of every learner. This outsiders' concept seems to have the potential to outcompete traditional normative pedagogy teaching models, in the sense of efficacy and effectiveness as depicted by learners' performance and satisfaction.

Keywords: Machine Learning; Model; Recommendations; Predictions;Adaptive learning;Sentiment Analysis; Content Based Filtering; Item-Based Collaborative Filtering; User-Based Collaborative Filtering

Acknowledgement

Firstly all credit goes to Almighty Allah for making the path easier and hinting the path during the crisis time.Then, Arif Shakil Sir, for taking me under his supervision. Also, thanks goes to my classmates who helped me in the survey and on other things.

Table of Contents

List of Figures

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

ϵ Epsilon

υ Upsilon

CBF Collaborative Filtering

CNF Contenr Based Filtering

HBM Hybrid Model

Chapter 1

Introduction

1.1

Technology has continued to enhance educational systems due to the advancements that enhance unique learning systems. Traditional educational paradigms are not often successful in addressing the diverse needs of students, who learn in different ways, at different pace, chance at different choices. In this thesis, I designed a Personalized Learning Platform Model Using Machine Learning in the hope of enhancing learning achievement through learner adaptation.

The aim of this project is to develop an ML-based learning environment for learning and assessment where the learner receives suggestions, learning guidance, or feedback that is based on a ML engine. Therefore, the platform has an aim to provide an individual and active learning environment using techniques such as: collaborative filtering, content based filtering and adaptive feedback. The method will eradicate common pitfalls usually seen in normal education delivery systems, for example, the system that considers all learners as though they have similar learning abilities and interest thereby producing half-baked learners.

In the first part of my thesis, I examine the concept of tailored learning and the contribution of machine learning to the enhancement of its approaches. The platform is meant to offer on-line updates of material delivery based on the learner's progress, making it much more focused on students.The second part of the thesis examines the concept and instantiating of the platform. This work uses learning techniques such as collaborative filtering, content-based filtering, and sentiment analysis to suggest learning resources that will be suitable for each student. In addition, the platform provides immediate feedback; it enables them to concentrate on the difficulty level of comprehending and producing content, as well as their accomplishments.

Extensive testing and assessments for utilization of individualist instructional approaches reveal that the method can render superior improvements in learning and students' satisfaction than traditional methods. This research adds to the emergent scholarship of educational technology by showing how machine learning can be applied to provide individualized, effective, and stimulating learning processes.

This thesis is structured as follows: the following chapter presents a review of the literature on personalized learning and the use of machine learning in education. Subsequently, key facets of the overall proposed approach and design of the custom learning platform, as well as the machine learning techniques employed by the platform, are expounded. Last of all, the findings and discussions from this study are stated, alongside with the potential future research under the context of learning systems personalization.

Recommendation systems means that the decision making process is simplified based on suggestions of proper items to use. On the personal level it can assist with shopping for necessities, movies and music, eating out, travel, and reading news. In our study, the term "items' is employed with reference to tutorial videos. Hence, joining the group of information filtering systems, recommended systems are a sub type of such systems.

1.2 Aims and Objectives

The goal of this work is to design an easy to use adaptive system for learning for each student specifically. It will utilize adaptive blended instruction technology to deliver recommendations concerning goals, interests, and history of the user. The system will give computerized feedback on individual needs and achievement and also self paced learning and revision with content that will change depending on the learner's understanding of the material and progress in real time. The hope of this technique is to address motivation, participation and academic achievement among students through individual approach.

1.3 Motivation :

Bangladeshi students know how to seek help from private tutors and coaching centers apart from receiving academic counseling from teachers of schools and colleges. Independent study becomes tricky for learners since they attend three credit hours from lectures each week for each course as they seek tertiary education. Besides class lectures and professors' directions, students often need more recommendations to understand new material more completely.As a consequence, learners address internet sources, especially different e-learning portals, to gain knowledge. It might take a while to find appropriate instructional websites.

It has been established that YouTube is the most preferred e-learning platform by the students in learning due to its ease to use and the AIM free as distinguished from other websites that needed membership. Students often use the YouTube channel for videos that give the visuals and even the voice over to make their learning process easier. A video tutorial is less time-consuming as compared with reading e-book, articles, or even presentation slides. Since students join it from different countries, you get videos in the local and other international languages. From a large pool of tutorials students have a chance to choose the videos they like.

YouTube's effectiveness is limited by its lack of a specific instructional platform, resulting in recommendations of irrelevant content such as movies, songs, and news.

Students may become distracted while browsing recommended videos for study resources, requiring additional time and effort. As undergraduate students, we believe a website with recommendation systems for instructive video lectures would be quite beneficial. As a result, this was the primary motivation for our research.

1.4 Problem Statement :

The strategies of instruction that have commonly been used might not help all students; especially in terms of speed and pace. This approach can lead to student disinterest, poor academic performance, and low level motivation. Specificas are favorable when it comes to learning because they meet each institution's needs of specific learners. Existing systems of personalized learning lack the capability to offer relevant feedback to learners as well as the ability to make suggestions on content relevant to any particular learner.

The present current feedback and suggestion system are wrong and counterproductive, which is a big problem. It is unlikely that the common strategies of recommending a list of items, relevant to the user's profile, will correspond to each learner with the best fit of the course content, learning history, and interaction patterns. However, feedback systems are not always individualistic with meta-cognition and cannot offer or advise pupils as needed.This project will develop a learning platform that cataylor the student's learning profile and will incorporate improved machine learning algorithms for better suggestions and feedback. The aim is to provide a system that will respond to the needs of each learner, making satisfaction higher and learning outcomes more successful.

Key issues to solve :

1. Inaccurate Recommendations
2. Generic Feedback
3. This way of teaching allow for the modification according to the pace that learner sets while studying.

This work seeks to solve the above challenges by developing a new learning model that will enhance a student's experience by harnessing machine learning algorithms.

Chapter 2

Literature Work

Before doing a project, make it quite praiseworthy a research on literature review on it's past history and evolution is must. Therefore, I have gone through some of the papers below:

2.0.1 Background :

It has been found that most of the recommendation systems can most often be built using one of It can again be divided into: Content-based filtering or, Collaborative filtering (CF) [5].Content-based filtering algorithm suggests products similar to those that every user was using before valued, utilizing the analysis of the content of the objects that the user has identified in the past [14].

The basis of recommendation system includes data mining, information retrieval, statistics, etc Recommendation systems are valuable as an option to search engine algorithms as these enable a user discover new things he would not otherwise have come across.Leskovec et al in the book [7] stated that in 1990, Jussi Karl gren, at Columbia University, Coined the Term "Recommender System" Math course, computer science course, He later on made successful large-scale deployment of ''Recommender System" He also remained actively involved with the same through different.

Existing recommendation algorithms often have difficulties identifying relevant content and providing the most relevant material more often than not learning materials for learners at individual levels; Thus in the case of significant considering the augmentation of both the number of users and resources in learning environments, recommendation.**ElAissaoui2019**There are going to be a lot of scaling problems that simple algorithms are going to run into, they will exceed what is feasible or acceptable.The use of recommender systems leads to computing limits [[6]], and they provide support to different user activities.In some application scenarios.It is expected that integrating several recommendation.Alone, these systems will produce better results as opposed to combining each approach of the identified strategies.[1].

Personalization was not an alien concept many years before the arrival of the term 'Recommendation Systems' was invented. Traditional recommendation systems appeared not long ago, and before that, there were no systems at all. people were doing personalization in a manual mode[4]. For instance a buyer can contract a salesman to.The options have to be selected and recommended manually and the dresses that would suit him would have to be identified.of the clothing. To regain the personal aspect into more extensive fields, the technology must first be applied.The recommendation systems where the concept of technological personalization was initiated were launched.The principle of recommendation systems is grounded on data mining and information.search, statistics and etc Recommended systems are the efficient substitute to the search algorithms.any other since the algorithms assist users in finding products they might otherwise not have come across.Citing from the book [8], Leskovec et al proposed, in 1990 at the Columbia University, Jussi Karl-Gren, for the first time used this terminology 'Recommender System' via one of.-tems' was invented. In the past, before recommendation systems had emerged,people performed personalization manually. Like, a buyer can engage a salesman to manually choose and suggest suitable dresses according to his choice or requirements of the clothing. In order to implement personalization into broader aspects, technological personalization was introduced in the form of the recommendation systems[3].

Hybrids generally work in a way that integrates at least two input data streams or, using several recommendation techniques, such as the collaborative filtering approach. The content-based method proposed by Li et al. [8] effectively delivered content related item sets utilizing collaborative filtering (CF).and then used the above formed item sets for the purpose of pattern mining which is subsequent. As a result, they was able to develop sequential pattern suggestions for learners.Indeed, TELrecommender systems are similar to those used in other fields, that is,to e-commerce, in terms of recommendation aims and targets, as well as users' roles and duties.When considered within the context of technology enhanced learning (TEL), the complexity of the tex(parameters are realized concurrently as tightly interlinked pedagogical.Thus, the theories and methodologies impacts recommendation [[11]].Inaccuracy may result from such issues as lack of data, the cold-start issue, or poor modeling of learner preferences; as a consequence, the levels of learner engagement and efficiency is be diminished [[7]].

The basic principle of recommendation systems is derived from the concepts of data mining, Information Retrieval, Stastical analysis etc Recommended system are the beneficial substitute for search algorithm because these assist the user to find the items that they never looked for. Interestingly, about the use of Recommender Systems, Leskovec et al in the book [9] pointed out that in 1990 at Columbia University, Jussi Karl-gren was the first to use the term "Recommender System '' inform of one of his technical reports . He then attempted and formalized large-scale application of his 'Suggesting Ommender System' and also continued the work of that through, "It was invented through different.items'[2]. Traditionally, before actual recommendation systems started appearing, people did personalization by themselves. Such as, a buyer can employ a salesman who will individually select and recommend preferable dresses depending on the buyer's preferences or need of the clothing. To incorporate personalization within larger spheres, the concept of technological personalization was developed as recommendation systems[12].

2.0.2 Related Work :

Since ML and data analytics have become effective ways of enhancing learning performances, the design of individualized learning environments has gained considerable interest in the e-learning domain. Some prior studies have addressed how machine learning can enhance the individualisation of contexts of e-learning by adapting contents and even the learners' interactions.

Supervised and unssupervised learning approaches were proposed by El Aissaoui et al., 2019 in an adaptive e-learning system based on the group learning styles. They employed the Felder-Silverman learning style model (FSLSM) to categorize learners by their behavior and preference, this way; the accuracy of materials recommended was improved [10].

One of the most concerns in e-learning programs is undoubtedly student dropout. The decision-tree, ANN, and Bayesian network models proposed by Mingjie Tan and Peiji Shao [10] state that it becomes easy for the administrators to identify the at-risk kids and prevent student dropouts by using machine learning approaches.

To understand challenges and opportunities in applying machine learning and data analytics to e-learning systems, Moubayed et al. [5]. They also realized the importance of data acquisition and utilization in identifying behavioral patterns of students as well as improving system performance. Thus, when classification and clustering models are applied into e-learning systems, technologies can offer more personalized information assurance to learners; enhancing learning activities and interactions in general.Aher and Lobo [11]studied the use of machine learning methods like K-means clustering and Apriori to choose the courses in e-learning systems depended upon the previous records of the students. The outcomes of their study are revealing the fact that clustering and association rule learning can be useful and effective to provide learners with enhanced course suggestions by tuning in to their previous experience and the preferred choices of other learners.

Chowdery and Rao [8] analysed agent based recommendation systems where specific focus was given to NLP and machine learning in enhancing the contents recommendation in an e-learning context. In their research, they suggest that the combination of NLP with clustering algorithms may improve on course recommendations and presumably customer satisfaction and achieved 98% accuracy in prediction.Chaudhary and Gupta proposed an intelligent web-searching model for e-learners based on the machine learning approach involving Random Forest and Support Vector Machines for the anticipation of user preferences for instructional content.

Rasheed and Wahid [13] found that learning styles need to be captured in e-learning systems in order to enhance the possibility of students' retention and understanding. For their study, they used several types of machine learning, including decision trees and neural networks, to categorize learners according to their learning preference profiles.Liu and Ardakani (2022) put forward an effective e-learning system model, wherein it applied machine learning strategies to adjust the learning material according to students' emotions. Their approach used EEG in capturing brain waves data and used K-Nearest Neighbors (KNN) to determine emotions.According to the study conducted by Zine et al. (2023), machine learning was applied to determine e-learning readiness. In the course of the study, Critical elements quantifying e-learning readiness were identified by applying Random Forest as well as Decision Tree algorithms at two levels which include; Ability and Knowledge.

Chapter 3

Dataset

The dataset used in this project is a primary dataset. The dataset consists of two main components: information on user interactions and details about item content. Basically I made 2 datasets from the form I shared to my classmates and others. One dataset was the content of the course and another one was about the interaction of the users.

3.1 Data Collection

Basic Statistics: The dataset contains a mix of unique and repeating values in different columns, such as course id (unique for each course) and subject (limited to a few categories). Numerical columns like price, num of subscribers, and num of reviews show significant variation, indicating diverse user engagement and pricing strategies. There are :

- Id
- Title
- Url
- Paid
- Price
- Subscribers
- Reviews
- Content duration
- Published times
- Subject
- Published date
- Published time: The time of day when the course was published.

For the first part of the research we collected data through surveys from the undergraduate students in the Computer Science and Engineering department of BRAC University in Bangladesh. Google forms and interviews were utilized in the administration of the survey. The final totals from all the surveys were 65. Soon, we ceased collecting more responses from the classmates and friends because the responses they provided were almost similar to each other.

A total of three students were asked to perform a second part of the study where they have to rate YouTube video tutorials for computer science classes and give their demographic data. Finally their rating about the full film was recorded. Instead of a Web Crawler, the parameters for the YouTube video were gathered manually due to the fact that it would have taken a considerable amount of time to obtain a few required parameters from a plethora of information which we had gone through. The former was followed by the implementation of the content-based filtering method, user-based collaborative filtering method, and item-based collaborative filtering method.

3.2 Data Loading and Pre-Processing :

3.2.1 Loading the dataset:

```
import pandas as pd
import numpy as np

# Load the dataset
df = pd.read_csv('course_data.csv')
```

Figure 3.1: Dataset loading

Handling the missing values:

Fill missing values in numeric columns with the mean of each column.

```
df['price'].fillna(df['price'].mean(), inplace=True)
df['num_subscribers'].fillna(df['num_subscribers'].mean(), inplace=True)
df['num_reviews'].fillna(df['num_reviews'].mean(), inplace=True)
df['num_lectures'].fillna(df['num_lectures'].mean(), inplace=True)
```

Figure 3.2: Missing Values

Converting Data types:

Convert columns to appropriate data types.

Saving Cleaned Data:

Save the cleaned DataFrame to a new CSV file

Flowchart of assessment

```
df['course_id'] = df['course_id'].astype(int)
df['price'] = df['price'].astype(int)
df['num_subscribers'] = df['num_subscribers'].astype(int)
df['num_reviews'] = df['num_reviews'].astype(int)
df['num_lectures'] = df['num_lectures'].astype(int)
df['is_paid'] = df['is_paid'].astype(bool)
```

Figure 3.3: Columns to datasets

```
categorical_columns = ['course_title', 'url', 'level', 'subject']
for col in categorical_columns:
    df[col] = df[col].astype('category')
```

Figure 3.4: Updating

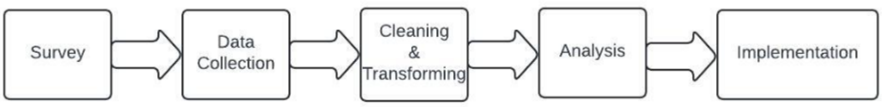

Figure 3.5: Assesment

Chapter 4

Methodology

In this part, I have worked with Content Based and Collaborative based filtering first. Then using the weighted method integrated both of them in a hybrid model.

Content Based Filtering :

It is a method that makes recommendations based on the properties of the items in this case, courses. This model analyzes the content of the courses such as the course titles and categories and compares them to the interests of the user.
From the view of Das [15], content based filtering is filtering technique that alerts the consumer regarding its recommended products and the products that have features related to products which the consumer has rated highly. The second is familiar to programmers and consists in that every object has an item profile, which is a table containing the item's attributes. The items, in this case, would be the course code, course topic, video title, language, length, views, likes, and dislikes. Indices are obtained and features are compared. So, on the basis of best matching scores, the system will suggest the goods that will be closer in nature to the user's requirement. It focuses only on the item attributes rather than the user.

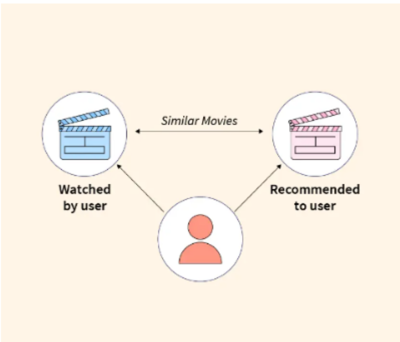

Figure 4.1: Con.BF,BotPenguin (2024)

Term Frequency-Inverse Document Frequency (TF-IDF):

TF-IDF stands for Term Frequency –Inverse Document Frequency is a metric used for calculating the weight or message about the importance of a particular word in the context of a particular document in reference to the other documents in the collection or corpus. Such as used in information retrieval and text mining for the transformation of the text into numerical values where significant terms are assigned high importance.

$$TF(t,d) = \frac{f_{(t,d)}}{\sum_{t' \in d} f_{(t',d)}}$$

Inverse Document Frequency (IDF):

The relevance of a term throughout the entire corpus is measured by the Inverse Document Frequency (IDF). Words that crop up frequently in papers have a lower weight since they are thought to provide less information. For term t, the IDF is provided by:

$$IDF(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Cosine Similarity:

When comparing two non-zero vectors in a multi-dimensional space, cosine similarity is a frequently used metric in information retrieval that is frequently used to analyze text or user-item interactions. In high-dimensional spaces, where the direction of the vectors is more significant than their size, it is very helpful. The cosine similarity coefficient yields a number between -1 and 1, where 1 denotes vectors that are identical. It calculates the cosine of the angle between two vectors.

$$\frac{\{A \cdot B\}}{\{\|A\|\|B\|\}}$$

Collaborative Filtering:

Explores user interaction data such as ratings, clicks, and purchase history to reveal patterns in user behavior and preferences. This approach is highly effective in capturing the underlying factors that impact user decision-making.

User Based Collaborative Filtering:

This cooperative filtering technique employs user to user connections. According to the choices made by a set of certain users, another set of users, which is similar to the one making the recommendation, is obtained.Using Pinela's ideas [20], the people with the similar coefficients or with the similar preferences are united, and the 11 goods with the best preferences are offered to the next user who has similar preferences. Based on the Work of Huang [27], user opinions could be categorized as explicit or implicit. The user's outcome is represented through the use of explicit opinion where his/her rating of an item (for instance an app or a movie) is identified as soon as the user completes rating it. Heuristics touching a user's likes or the preferred characteristics of an item are drawn from implicit opinion.

Item-Based Collaborative Filtering Method :

Item-item connection is the technique applied in this decision making of using collaborative ltering method. For a particular An item is also found out, when other

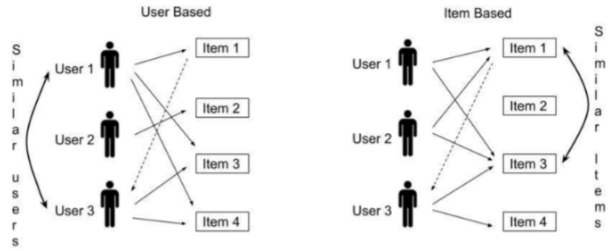

Figure 4.2: CBF, Upadhyaya, Prakash. (2023).

items which have the same features and characteristics are identified. The rating for the item iscalculated using the estimation which is derived as the similarity between the similar bunch of items. Tomarin [24] stated, for calculation of similarity between two items, the set of items the This vector also takes into consideration target user has rated and how close they are to the target item i.computed and k most similar to it items are chosen. To nd the similarity of products; all the users are also taken into consideration in addition to the target user. Also it was written by Lew et al. [29], it is possible to define similarity between two items of a hierarchy. the ratings which have been given by the users who have evaluated both the items. Then, adjusted cosine Many configurations of the nodes are possible and a similarity measure is used for calculation.The items that appear to be more similar are. It is calculated using the given formula :

$$P_{u,i} = \frac{\sum\limits_{\text{all smaller items},N} (S_{i,N} \cdot R_{u,N})}{\sum\limits_{\text{all similar items},N} (|S_{i,N}|)}$$

Whole project's design :

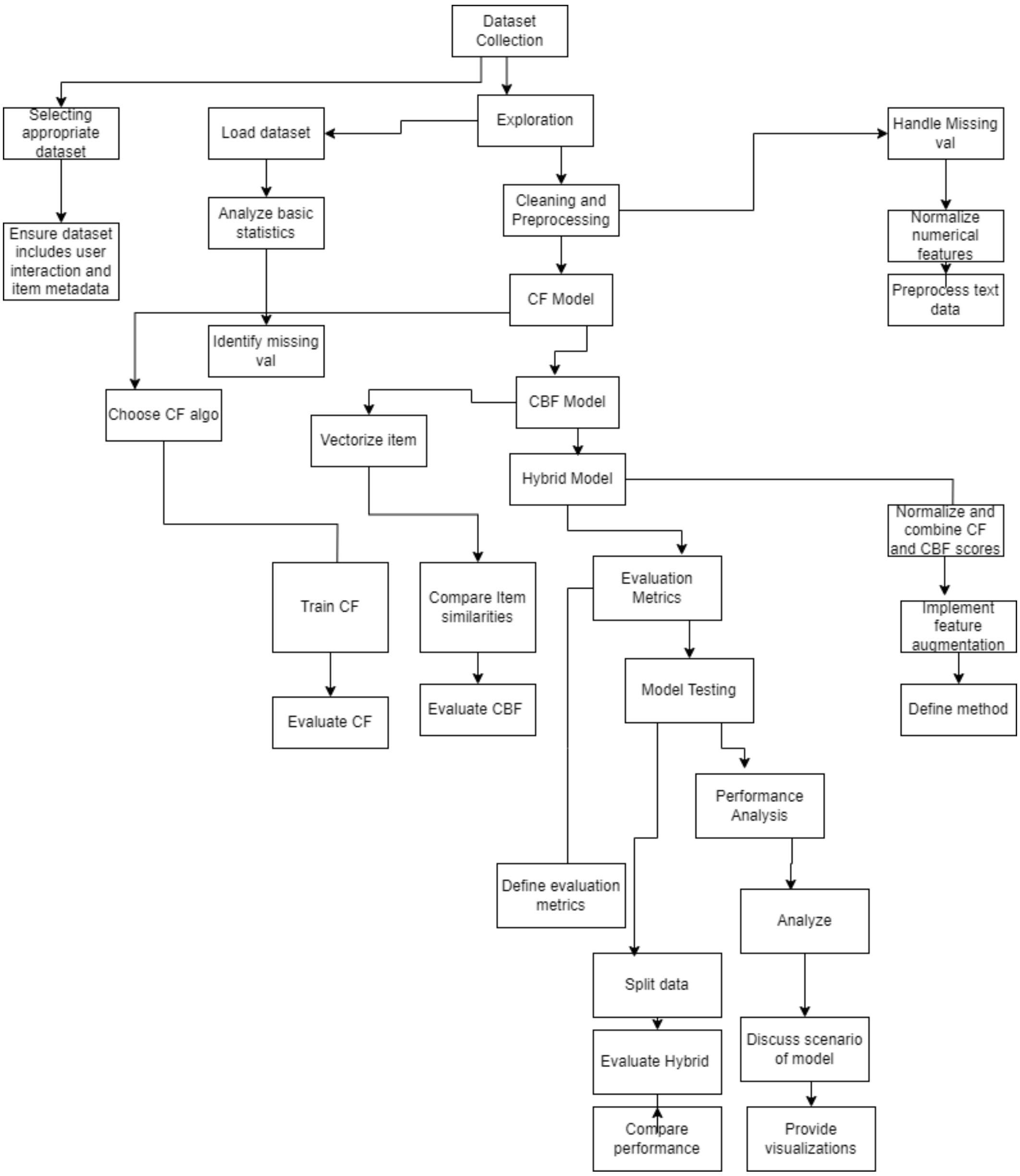

Figure 4.3: Architecture of the model

Chapter 5

Implementation

5.1 Content Based Filtering :

We then linearly transformed the course titles into numerical vectors using the TF-IDF vectorization.To determine the extent to which courses titles are similar, we calculate the cosine similarity of these vectors.A user selects a categorize certain courses (such as "Python" or "Data Science"), and the system recommends courses with content similar to those courses with which the user has interacted.

Advantages :

As it is not authorized to compose anything except the course content, it is very useful for new users who may have little participation earlier.These suggestions can be explained since they are made on the basis of the nature of the interested courses.

Limitations :

If there is not much difference in the content being taught or where the titles do not reflect the content of course being offered, then probably it may not work perfectly well. However, the degree of appropriateness is lower when the user has many diverse interests as compared to collaborative techniques.

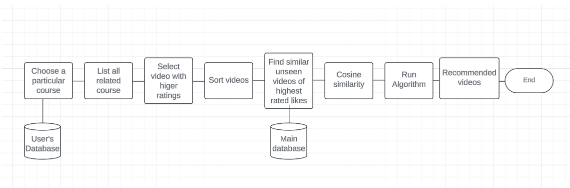

Figure 5.1: Workprocess of Content Based Filtering

5.2 Collaborative Filtering :

The use data, or the courses that users have engaged through view, like, or enroll, is looked at during collaborative filtering. As stated earlier we look for similar people from the shared courses using interactions. Based on these findings, we can

recommend the following one; presenting courses that user A has seen but user B has not if only they have seen many similar courses.
Using the obtained algorithm, individuals with similar preferences can be recommended to similar courses that are popular among users with similar interests.

Advantages :

It is highly adaptive to user desires, as opposed to the course content users engage in; and it provides recommendations tailored to the user's actions. It is a recommendation method that gives a broad range of recommendations as compared to a certain category or course and can better represent the users' diverse preferences.

Limitations:

More so, it is very much used to the user preferences as it makes recommendations based on behavior not class content. That way, the approach is rather universal with recommendations that may not even be tied to a specific category or course.

Cold-start problem:

Brand-new users may not benefit from collaborative filtering as it requires a sufficient number of samples of user interaction to make reliable recommendations because low-activity or short life-span courses may have little interaction data. It is based on sufficient information as well. The recommendations could not be precise or varied if the applications have no user or course interaction data to analyze.

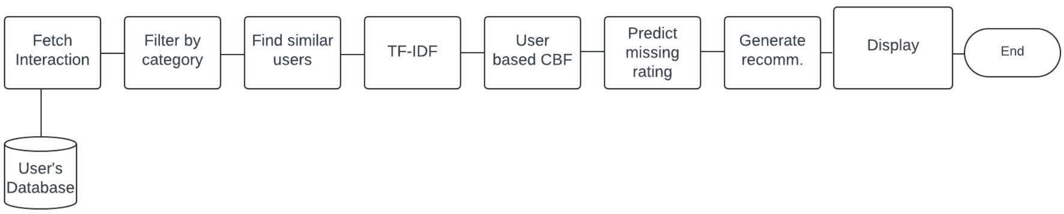

Figure 5.2: Workprocess of CBF

5.3 Hybrid Model:

Recommendations that are more evenly distributed thanks to a blend of collaborative and content-based filtering.Depending on the user's past interactions, the hybrid model dynamically modifies the relative weight of each strategy (collaborative or content-based). For instance, the system prioritizes collaborative filtering if the user engages in a lot of interactions, whereas it depends more on content-based filtering if the user engages in few interactions.

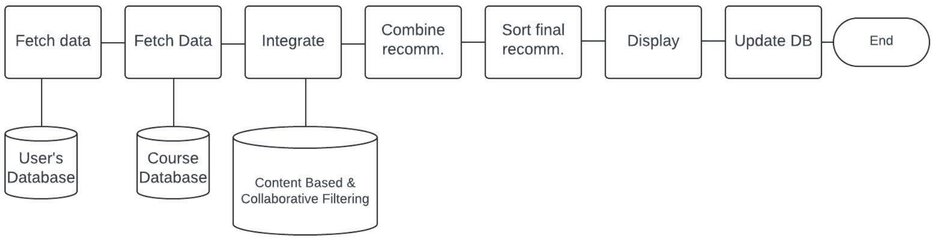

Figure 5.3: Workprocess of Hybrid Model

Chapter 6

Integration

6.1 Backend :

FastApi : FastAPI is a modern web frame for creating APIs with Python 3.6+ and it uses normal Python type hints for plumbing. It is asynchronous, fast – that is what the name suggests, and it supports OpenAPI and Swagger UI for generating interactive API documentation.

Application to the Project:

API Endpoints: Based on FastAPI, API endpoints for course loading, quiz creation, recommendation request, and assessment have been prescribed.
One of the primary reasons for choosing FastAPI is the capability to handle multiple user requests simultaneously because of the asynchronous conditions of the framework.

Automated Documentation: Why FastAPI – it has an active development community, and developing with it, an interactive SwaggerUI is generated automatically provided with which you can test each endpoint.

Benefits:
Due to use of asynchronous queries, optimization is achieved hence high performance. All sources and support for data conversion and validation are provided in. Works in a non-code way and produces documentation that is compliant with OpenAPI.

```python
# Step 3: Define endpoint to retrieve all courses
@app.get("/courses")
async def get_courses():
    courses = list(courses_collection.find({}, {"_id": 0}))
    return JSONResponse(content=courses)
```

Figure 6.1: Flask

MongoDB (NoSQL Database):
Data is stored in MongoDB, a document-oriented NoSQL database, with its data in a flexible JSON-like document. Such applications are said to boast of their ability

to deal with unstructured information and of their horizontal as well as vertical scalability.
Application to the Project:
MongoDB is used to store all the basic details of the course along with the user data, titles, categories etc.

User Interaction Data: For cooperative filtering of recommendation, it additionally retains user interaction information (like courses watched or completed).
The Reason It Was Used:
So, as long as the project is still in the making, MongoDB allows for changes in the user and course data structures with ease. The horizontal scalability characteristic of the platform allows for its growth as more datasets and more users are accommodated.

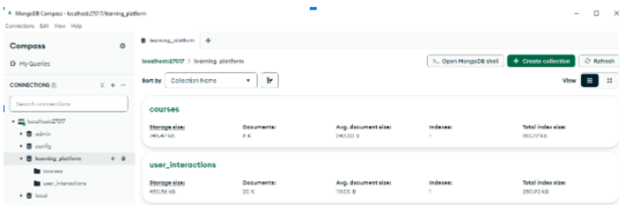

Figure 6.2: Mongodb connection

OpenAI GPT-3.5 Turbo Model :

Goal: The test questions are automatically derived from the contents of courses where GPT-3.5 is employed for creating the questions. It reckons utilization of OpenAI.Finalization.Create new content based on a given prompt through the create() API.
Benefits:
Natural Language Generation: Indeed, using GPT-3.5, it is possible to generate high quality natural language outputs, which enables quiz authors to create correct and interesting questions or statements on their quizzes.

API Flexibility: Accessible through API, OpenAI can easily be incorporated into the FastAPI backend to create quizzes automatically from the contents taught in the course. GPT-3.5 is very useful when the user requires real-time dynamic content creation and a fixed list of the quiz questions does not exist.

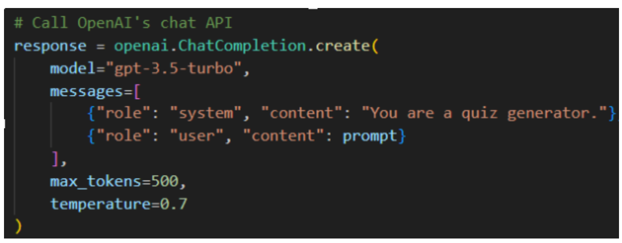

Figure 6.3: Using GPT Api

Swagger User Interface:

Swagger UI is an open source tool that has been integrated into the FastAPI application. An interactive API documentation page for testing your application's endpoints is created as a byproduct based on the endpoints in your application. In this project, other API endpoints including a course search, quiz generation, and recommendation systems are illustrated and used in Swagger UI .

The goal of Swagger UI :

Interactive Documentation: Using Swagger UI, developers or testers, just as with soapUI, need not develop more client-side code and can navigate the API by studying and using it.

Testing APIs: The ease of Live API calls means that data can be added into the Swagger UI forms (for example; user$_idorcategoryorquizanswers)andtheresultsinreal-timeacteduponorexamined.$

Endpoint Visualization: This makes the input/output format easy to comprehend just by automatically listing all the API endpoints side by side with the expected details of the request and response.

Benefits:

Auto Generation: Just as a reminder, there is no more setup necessary for FastAPI since it will now auto-generate the Swagger documentation.

Live Interaction: This means that with some input entries in the required parameter boxes, one is able to interface with this API through a web browser, and the system will respond with error codes or success answers at the bottom of the page.

Easy to Use:

When developers and QA testers fill out the form fields and click the "Try it out" button to execute the API call, then they can easily test the API endpoints. Swagger UI in this project allows me to test:

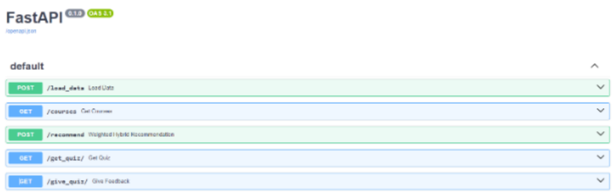

Figure 6.4: Using SwaggerUI

The role of Swagger UI in the Project:

Integration with FastAPI: Swagger UI is third party UI that opens immediately a FastAPI application is started. You can view the Swagger UI page with all of the routes listed by going to http://localhost:5000/docs.

Real-Time API Testing: Swagger UI enables one to test his APIs right from browser; there is no need to use programs such as Postman or test certain API related features.

Instruments for Integrating Swagger UI:

Natively, FastAPI addresses Swagger UI documentation.

Uvicorn: Runs the FastAPI server and it automatically opens the Docs page that is in the form of Swagger UI to interact.

6.2 Front-end:

React:

React.js is one of the most used JavaScript packages that is supported by Facebook, which is free to use to design UI patterns mainly for single page applications. In a way, due to this or that feature of component-based architecture, it can be possible to generate reusable UI components.

React provides complex UI updates, rendering with high reactivity by using Virtual DOM, and modular structure by using components.

Tailwind CSS: This utility-first CSS framework enables one to design differently without changing from HTML to CSS. It saves time for developing the software because it provides already ready utility classes like text-center, big-blue-500, m-4, among others.

In as much as the styling of individual components is concerned, Tailwind CSS is imported directly to the JSX or used in the App.css. That is most likely why your course recommendation system is both responsive and visually engaging.

6.2.1 Images of the Website :

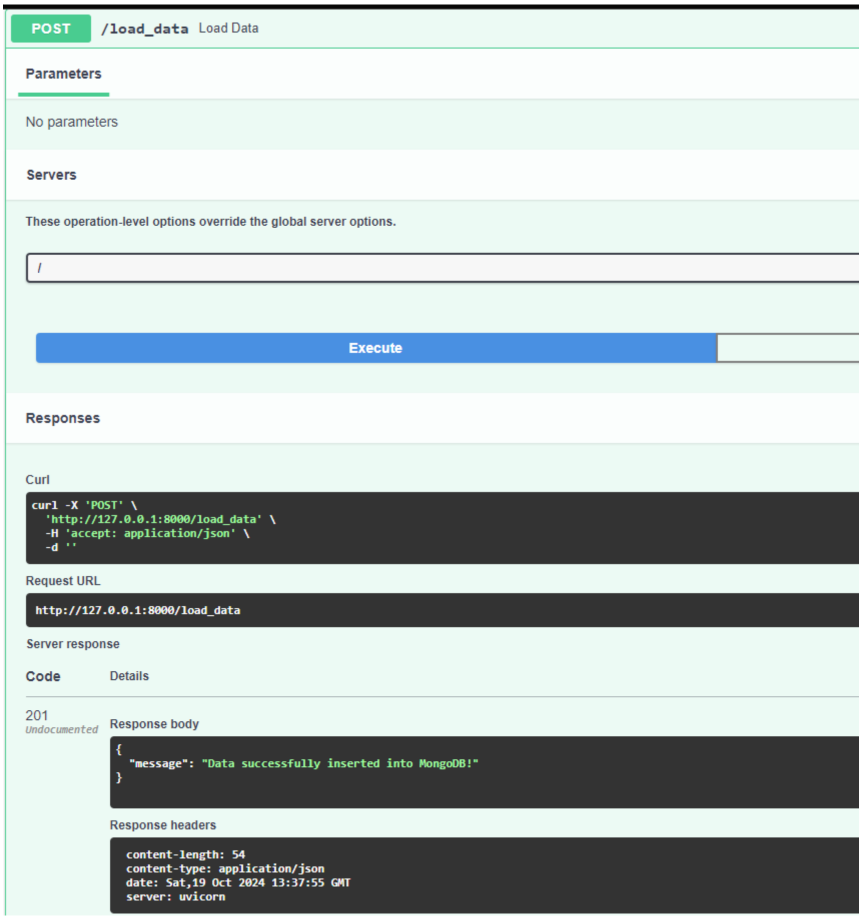

Figure 6.5: After loading data

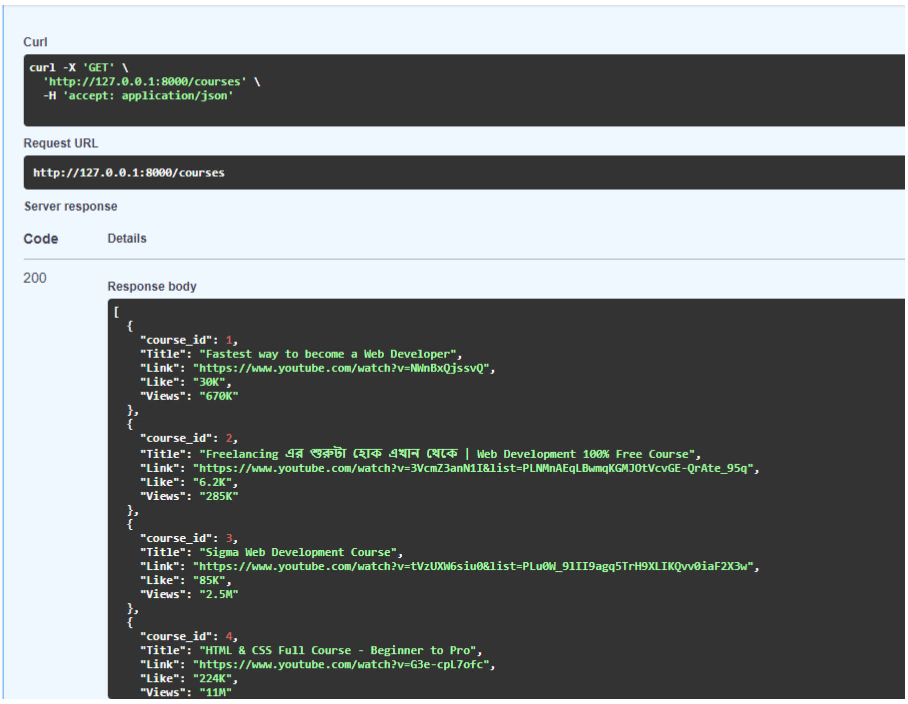

Figure 6.6: After getting all courses data

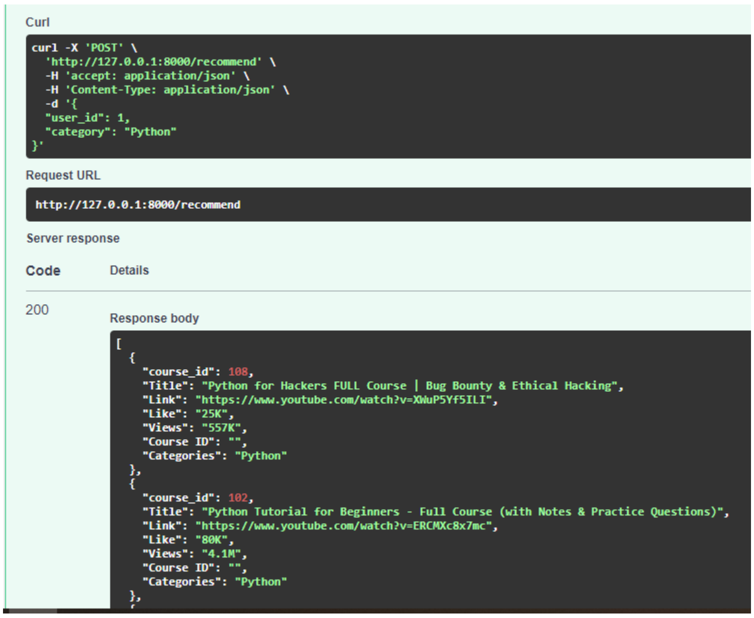

Figure 6.7: Weighted Recommendations of user

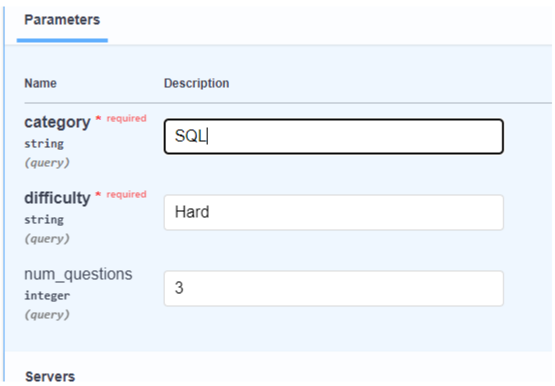

Figure 6.8: Quiz Prompt

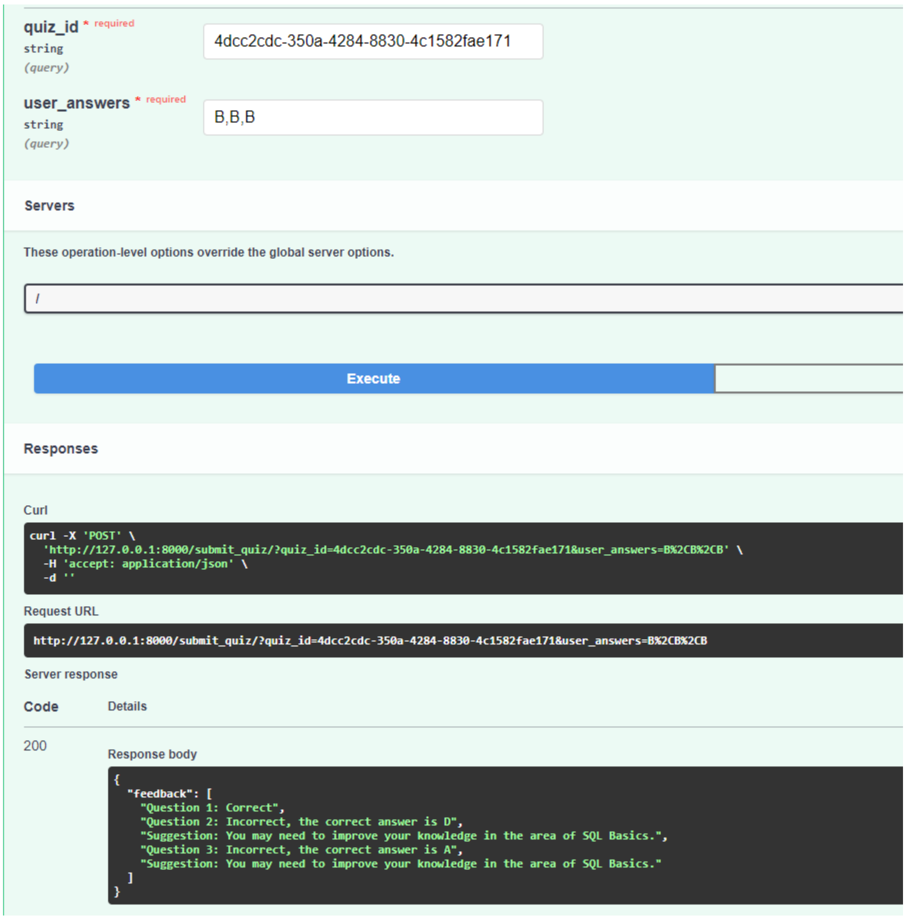

Figure 6.9: Feedback of quiz

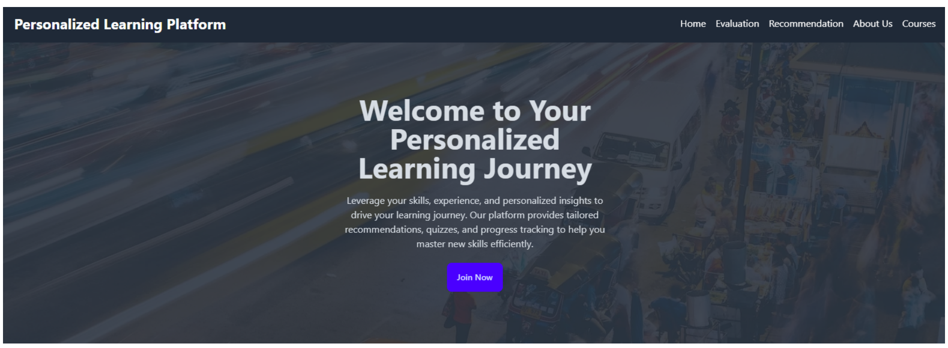

Figure 6.10: Home Page

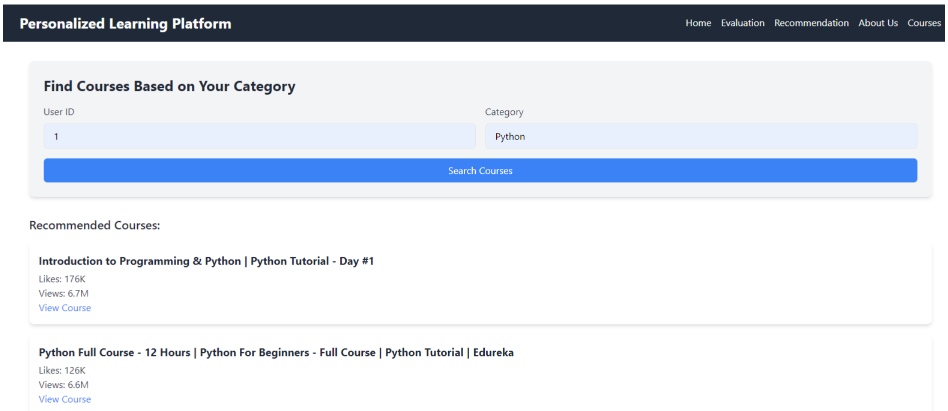

Figure 6.11: Weighted Recommendation

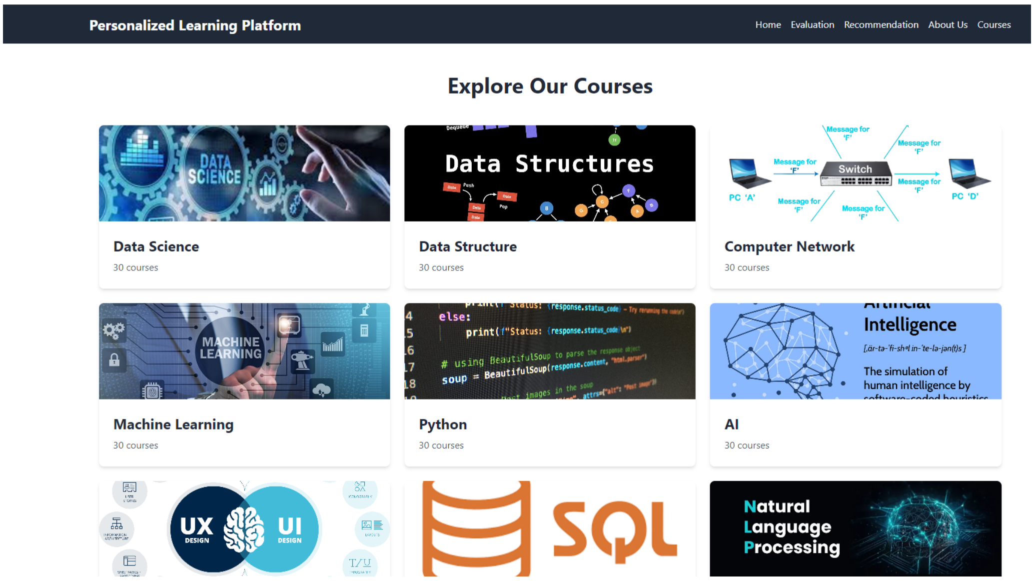

Figure 6.12: View Courses

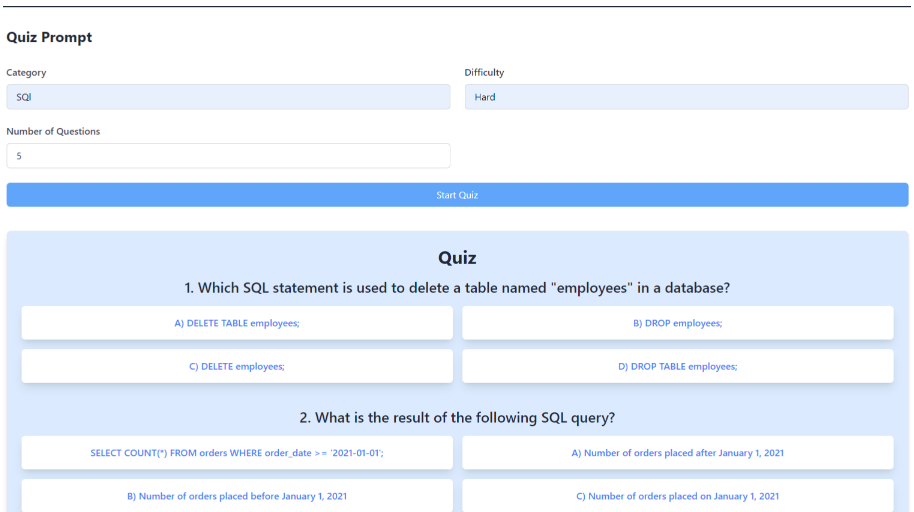

Figure 6.13: Evaluation

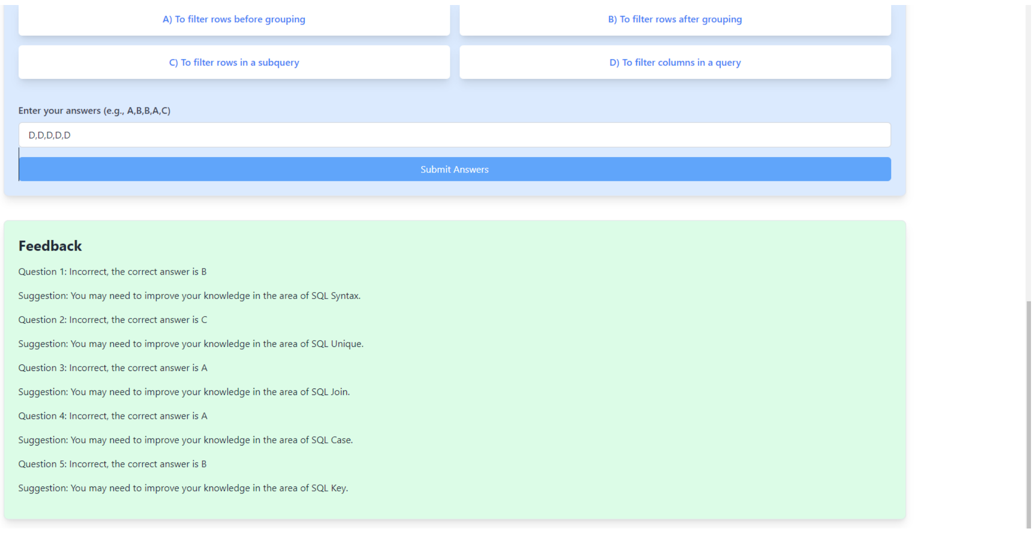

Figure 6.14: Feedback

Chapter 7

Future Works & Limitations

As there are a few limitations still in this model, we look further on it. In future, we may look on the hybrid model if it can work well with neural collaborative filtering or any other model. Then, the advanced website application is on the queue list for this project. Potential contextual factors include, mood, time of day, or incidents and events happening within the user's environment which may be used to arrive at more relevant recommendations. For example, a user may decide to attend shorter courses in the week and comprehensive courses at the weekend.

A better learning experience might be offered if the system was upgraded such that it can update recommendations as the user continues to enroll in more courses and consume more content. A system can always learn from the user behavior and adapt recommendation models using strategies like reinforcement learning as the user behavior may change over time.

The number and assurance of the questions might be improved by its expansion to integrate a large folder to organize the quiz creation feature. It would also be even more comprehensive if teachers or users could feed this bank of questions.It could also mean scaling the platform and keeping up with its maintenance should it grow by using a microservices architecture. That being said, the quiz creation module, the user interaction module and the recommendation engine would have to be separated into different services.By shifting the system to cloud using AWS or Azure the platform would be able to support more users and would have global access. This also makes it possible to scale out horizontally in order to meet new and growing needs sufficiently.

Chapter 8

Conclusion

Here we developed a platform using hybrid model which is a integation of content based filtering and collaborative filtering. It will try to give a recommendation to the user as a non biased as much as possible. Also, this platform featured individual learning methods and gives a suggestions to their weakness which ultimately help them to have a clear basic or foundation to the topic.

Apart from this, this system has some area where the tunings are needed. Probably, implementing another model to the current existing model will help to give a more accurate recommendation and a non biased outcome. Also, the course database may enrich day by day, and the quiz module can be improved by using higher api model architecture in future. Furthermore, the cloud based data storage system can help the overall functionality.

Lastly, it's an attempt to make an machine learning based platform which may help the students to boost their academic knowledge. With further development, this platform can help in a significant manner in the upcoming days.

Bibliography

[1] I. Bhandari, E. Colet, J. Parker, Z. Pines, R. Pratap, and K. Ramanujam, "Advanced scout: Data mining and knowledge discovery in nba data," *Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 121–125, Mar. 1997, ISSN: 1573-756X. DOI: 10.1023/A:1009782106822. [Online]. Available: https://doi.org/10.1023/A:1009782106822.

[2] M. Lee, P. Choi, and Y. Woo, "A hybrid recommender system combining collaborative filtering with neural network," in *Adaptive Hypermedia and Adaptive Web-Based Systems*, ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2002, pp. 531–534.

[3] S. Vembu and S. Baumann, "A self-organizing map based knowledge discovery for music recommendation systems," in *Computer Music Modeling and Retrieval*, ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2005, pp. 119–129.

[4] M. Bailey and S. Clarke, "Predicting the match outcome in one day international cricket matches, while the game is in progress," *Journal of sports science & medicine*, vol. 5, pp. 480–7, Dec. 2006.

[5] X. Su and T. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 2009, Article 4, 2009. [Online]. Available: http://www.hindawi.com/j.s/aai/2009/421425/.

[6] B. Darab and G. Montazer, "An eclectic model for assessing e-learning readiness in the iranian universities," *Computers & Education*, vol. 56, pp. 900–910, 2011. DOI: 10.1016/j.compedu.2010.11.002.

[7] S. Akhtar and P. Scarf, "Forecasting test cricket match outcomes in play," *International Journal of Forecasting*, vol. 28, Jul. 2012. DOI: 10.1016/j.ijforecast.2011.08.005.

[8] G. Gartheeban and J. Guttag, "A data-driven method for in-game decision making in mlb: When to pull a starting pitcher," Aug. 2013, pp. 973–979. DOI: 10.1145/2487575.2487660.

[9] T. Hopmans. "A recommendation system for blogs: Setting up the prerequisites (part 1)." [Online]. Available: https://www.themarketingtechnologist.co/building-a-recommendation-engine-for-geek-setting-up-the-prerequisites-13/. (Aug. 2015), [Online]. Available: https://www.themarketingtechnologist.co/building-a-recommendation-engine-for-geek-setting-up-the-prerequisites-13/.

[10] M. Tan and P. Shao, "Prediction of student dropout in e-learning program through the use of machine learning method," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 10, no. 1, pp. 11–17, 2015. DOI: 10.3991/ijet.v10i1.4189. [Online]. Available: https://doi.org/10.3991/ijet.v10i1.4189.

[11] F. E. Gunawan, A. Maryanto, Y. Udjaja, S. Candra, and B. Soewito, "Improvement of e-learning quality by means of a recommendation system," in *2016 11th International Conference on Knowledge, Information and Creativity Support Systems (KICSS)*, 2016, pp. 1–4. DOI: 10.1109/KICSS.2016.7951421.

[12] M. Maravanyika, N. Dlodlo, and N. Jere, "An adaptive recommender-system based framework for personalised teaching and learning on e-learning platforms," in *2017 IST-Africa Week Conference (IST-Africa)*, 2017, pp. 1–9. DOI: 10.23919/ISTAFRICA.2017.8102297.

[13] M. Ezz and A. Elshenawy, "Adaptive recommendation system using machine learning algorithms for predicting student's best academic program," *Education and Information Technologies*, vol. 25, no. 4, pp. 2911–2929, 2019. DOI: 10.1007/s10639-019-10049-7. [Online]. Available: https://doi.org/10.1007/s10639-019-10049-7.

[14] A. Yassine, L. Mohamed, and M. A. Achhab, "Intelligent recommender system based on unsupervised machine learning and demographic attributes," *Simulation Modelling Practice and Theory*, vol. 107, p. 102 198, 2021. DOI: 10.1016/j.simpat.2020.102198.